LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

UCRL-CONF-231266

Model-based Layer Estimation using a Hybrid Genetic/Gradient Search Optimization Algorithm

D. Chambers, S. Lehman, F. Dowla

May 25, 2007

Adaptive Sensor Array Processing Workshop
Boston, MA, United States
June 5, 2007 through June 6, 2007

Disclaimer

Model-based Layer Estimation using a Hybrid Genetic/Gradient Search Optimization Algorithm

David H. Chambers, Sean Lehman, Farid Dowla
Lawrence Livermore National Laboratory

Abstract: *A particle swarm optimization (PSO) algorithm is combined with a gradient search method in a model-based approach for extracting interface positions in a one-dimensional multilayer structure from acoustic or radar reflections. The basic approach is to predict the reflection measurement using a simulation of one-dimensional wave propagation in a multi-layer, evaluate the error between prediction and measurement, and then update the simulation parameters to minimize the error. Gradient search methods alone fail due to the number of local minima in the error surface close to the desired global minimum. The PSO approach avoids this problem by randomly sampling the region of the error surface around the global minimum, but at the cost of a large number of evaluations of the simulator. The hybrid approach uses the PSO at the beginning to locate the general area around the global minimum then switches to the gradient search method to zero in on it. Examples of the algorithm applied to the detection of interior walls of a building from reflected ultra-wideband radar signals are shown. Other possible applications are optical inspection of coatings and ultrasonic measurement of multilayer structures.*

A one-dimensional model-based reconstruction code was constructed using a simple analytical model for the radar reflection from a sequence of layers. The reflection model generated reflected pulses from a sequence of walls, then white noise added to create simulated data. An iterative inversion procedure is then used to back out wall positions, thicknesses, and material parameters from the simulated data. The wall estimation procedure begins with the wall closest to the transmitter/receiver. Time gating is used to select only the initial reflections to process for the first wall. After the first wall is determined, a later time gate is used to estimate the parameters of the second wall. By selecting later time gates, wall parameters are estimated in sequence from the outermost wall to the innermost. Using this progressive estimation approach, only a few parameters need to be estimated at a given time.

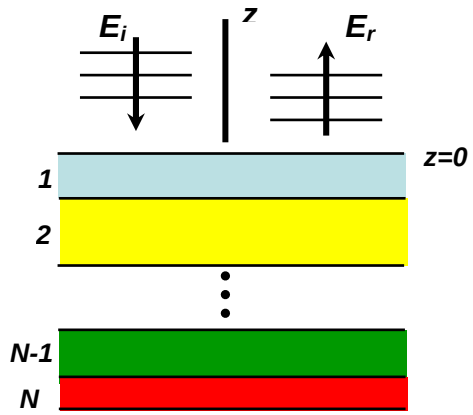

Figure 1: One-dimensional layer stack

The one-dimensional reflection model is based on the propagation model found in the text by Chew [1]. Consider a stack of N layers perpendicular to the z axis, the $z=0$ the top of the first layer (Figure 1). Given an electric field pulse $E_0(,t)$ launched from a point z_s above the surface, the reflected field at $z=z_s$ can be written as

$$E_r(z_s,t) = \int_{-\infty}^{\infty} \hat{E}_0(\omega)\, R_{eff}(\omega)\, e^{i\omega(t-2z_s/c_0)}\, d\omega\,, \quad (1)$$

where R_{eff} is the effective reflection coefficient for the stack of layers and $\hat{E}_0(\omega)$ is the Fourier transform of the transmitted pulse. The effective reflection coefficient is obtained by solving the backwards iterative equation

$$\tilde{R}_{j,j+1} = \frac{R_{j,j+1} + \tilde{R}_{j+1,j+2} X_{j+1}^2}{1 + R_{j,j+1}\tilde{R}_{j+1,j+2} X_{j+1}^2} \quad , \quad X_j = e^{-2i\omega\ell_j/c_j} \quad (2)$$

starting with $\tilde{R}_{N+1,N+2} = 0$, and ending at $R_{eff} = \tilde{R}_{0,1}$. The coefficients $R_{j,j+1}$ are the Fresnel reflection coefficients for each layer interface:

$$R_{j,j+1} = \frac{n_j - n_{j+1}}{n_j + n_{j+1}} \quad , \quad n_j = \frac{c_0}{c_j} = \sqrt{\varepsilon_j + \frac{i\sigma_j}{\varepsilon_0\omega}}\,. \quad (3)$$

Each layer is characterized by relative permittivity ε_j, conductivity σ_j, and thickness ℓ_j, with $n_0 = n_{N+1} = 1$. Figure 2 shows an example of three walls separated by 3 meters with a pulse emitted from a point one meter from the first wall. The walls are 10 cm thick concrete with $\varepsilon = 5.593$ and $\sigma/\varepsilon_0 = 0.0246$. The reflected pulse train in Figure 3 shows both direct returns from each wall and returns from multiple reflections between walls. The wall separations were chosen so that the direct reflection from the third wall would be

superposed on the return from multiple reflections between the first two walls, making the inversion for the third wall depend on the fidelity of the inversion of the first two walls.

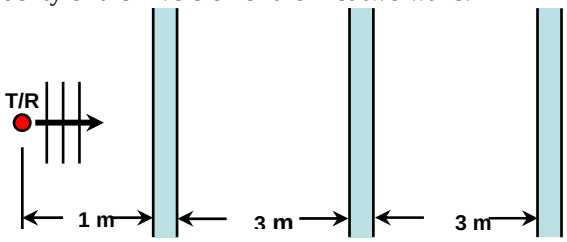

Figure 2: Three wall configuration and transmitted pulse (ns).

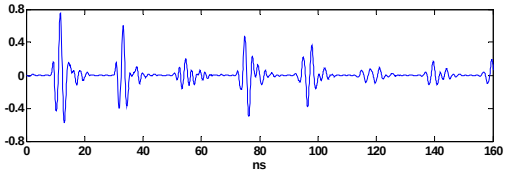

Figure 3: Reflected pulse train from three wall example.

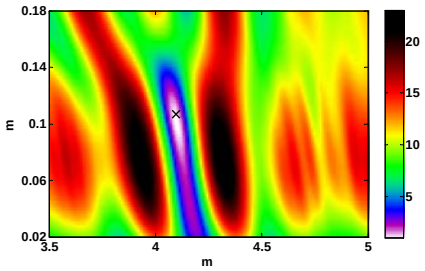

Figure 4: Error as a function of range (x –axis) and thickness (y-axis) for second wall. Reconstruction algorithm estimate of minimum is indicated by the "x".

The reconstruction algorithm estimates the range, thickness, permittivity, and conductivity of each wall in the model. Starting from an initial guess, it evaluates the forward simulator (Eq. 1) to obtain a predicted reflected return, compares it with the data within a specified time window, then calculates updated parameters. This procedure is repeated until the mean-squared error between predicted data E_{est} and actual data E_{dat} in the specified time window is minimized:

$$Q = \int_{t_s}^{t_f} \left| E_{dat}(t) - E_{est}(t) \right|^2 dt \,, \qquad (4)$$

where t_s and t_f are the start and end times for the time window. Figure 4 shows an image of Q as a function of wall range and thickness for the time window that includes the return from the second wall in Figure 3. The error has a prominent, well-defined minimum at the location of the true values for range and thickness. However, there are other local minima visible in this slice that would pose problems for any gradient descent minimization algorithm. One way to overcome this problem is to genetic algorithms or simulated annealing that randomly sample the error in the expected ranges of the parameters. Unfortunately, these algorithms can be computationally inefficient, making them difficult to use by themselves. After investigating simulated annealing and various ways to bootstrap a gradient search using various kinds of data preprocessing, we discovered that a combination genetic/gradient descent algorithm was most reliable. The particular genetic algorithm that we chose is the *particle swarm optimization* (PSO) method [2]. A Matlab implementation of this algorithm written by Brian Birge is available in the user-contributed functions area of the Mathworks website (http://www.mathworks.com/). This implements the *common PSO algorithm* which begins by choosing a set of random sample points $\left\{ x_i : i = 1, 2, ..., N \right\}$ within the parameter space. The error functional is calculated at these points. Displacement vectors v_i are calculated for each sample point (particle) using the following formula for the *kth* iteration:

$$v_i(k+1) = \phi(k) v(k) + \alpha_1 \gamma_{1i}(k) \left[p_i(k) - x_i(k) \right] + \alpha_2 \gamma_{2i}(k) \left[G(k) - x_i(k) \right] \,,$$

where $\phi(k)$ is a linearly decreasing inertia factor, α_1 and α_2 are acceleration constants, and $\gamma_{1i}(k)$, $\gamma_{2i}(k)$ are random number uniformly distributed in the interval between zero and one. The point $p_i(k)$ is the position with minimum error found by particle *i* on its trajectory through the *kth* iteration. The point *G(k)* is the current global minimum over all the particle trajectories up to iteration *k*. Once the updated displacement vectors are calculated, new sample points are determined by adding them to each sample point, $x_i(k+1) = x_i(k) + v_i(k+1)$. The error functional is evaluated at the new sample points and the process is repeated until all the particles converge on the global minimum. The convergence of the algorithm depends on the choice of acceleration constants, inertia factor, number of particles, and initial size of the parameter space. For our case we chose 24

particles for searching a four-dimensional parameter space for each wall.

After investigating the convergence behavior of the PSO algorithm, we discovered that the estimate of the global minimum found after only 20 iterations was close enough to the true global minimum that a gradient descent algorithm or a simplex could be used without encountering a local minimum. We implemented both methods, using the Matlab function fminunc for the simplex method and lsqcurvefit for the gradient descent Levenberg-Marquardt algorithm. These are found in the Matlab Optimization Toolbox. The gradient descent algorithm has the advantage of allowing the computation of the Cramer-Rao lower bounds of the variances of each parameter from an estimate of the noise variance in the data. The simplex method allows more general definitions of error *Q* that might be better if the noise is non-Gaussian.

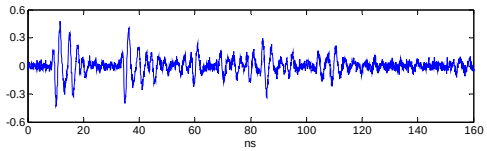

Figure 5: Simulated data for three thick walls, SNR = 20 dB.

As an example of the reconstruction process, consider the case of the three walls in Figure 2 except we change the wall thickness from 10 cm to 30 cm. The resulting reflected pulse with -20 dB of white noise added is shown in Figure 5. We start by estimating the parameters of the first wall using only the reflected pulse in the window between 5 and 30 ns. After 20 iterations of the PSO algorithm and 10 iterations of the gradient algorithm, we obtained the values found in the first column of Table I for the first wall. Next we select the time window between 30 and 55 ns, which includes the return from the second wall. Using our parameter estimates for the first wall, we search for the set of second wall parameters that minimizes the error between model and data (second column of Table I). Again, the algorithm starts with 20 iterations of the PSO algorithm and finishes with 9 iterations of the gradient method. Finally, we select the time window between 55 ns and 100 ns to estimate the third wall parameters, which converges after only 7 iterations of the gradient algorithm. The error bounds for each parameter are lower bounds determined from the data noise variance. We see that the range and thickness estimates agree quite well with the true values 0.3 m for thickness and 1.0 m, 4.3 m, and 7.6 m for ranges to each wall. Estimates of the relative permittivity and conductivity are less precise but still reasonable close to the true values of 5.593 and 0.0246.

Table I: Estimated parameters for 3 walls. Exact ranges are 1 m, 4.3 m, and 7.6 m. Wall thicknesses are 30 cm. Permittivity is 5.593, and $\sigma/\varepsilon_0 = 0.0246$.

	Wall 1	**Wall 2**	**Wall 3**
Range	0.999 ±0.0003	4.2963±0.0003	7.5878±0.0003
Thickness	0.299±0.002	0.291±0.002	0.294±0.001
Permittivity	5.65±0.06	5.95±0.06	5.84±0.05
σ/ε_0	0.031±0.003	0.035±0.003	0.024±0.023

To date we have investigated the sensitivity of the reconstruction algorithm to noise level, wall attenuation, and wall thickness. We have also investigated the case where the algorithm tries to estimate the parameters for a wall that is not actually present in the data. From the noise study we determined that we could get reasonable estimates of wall thickness and position for SNR greater than a few dB. This carried over to the study of highly attenuating walls. As long as the SNR for the return of a wall was greater than a few dB, the algorithm gave fairly accurate results for range and thickness. However, the estimates of permittivity and conductivity were more sensitive to SNR. Thus geometrical parameters for walls behind thick attenuating outer walls could be estimated as long as the returns from the interior walls were above the noise level. The material parameter estimates are less of a concern since they do not contribute directly to the extraction of the geometry.

The final studies of the one-dimensional MBT were Monte Carlo simulations designed to find a statistical indicator that could be used to test for a false wall. We used a two wall configuration where half the simulated data contained reflections from the inner wall and the other half only from the exterior wall. Each ensemble consisted of over 2000 data series, each member generated by a different realization of noise. For each wall, the MBT estimated its range, thickness, relative permittivity, and conductivity. Histograms of these parameters were compiled to determine which were good indicators if a wall was real or false. From this analysis, we found that the histograms of wall thickness and

permittivity differed noticeably between the true and false wall cases. Figure 6 shows a scatter plot of permittivity and thickness for the second wall when the first wall is composed of 10 cm thick concrete with low attenuation. We can combine permittivity and thickness by calculating the optical path difference (OPD) for the wall: $OPD = t_w(\sqrt{\varepsilon} - 1)$. Histograms of OPD show excellent separation between the real and false wall (Fig. 7). Figure 8 plots the probability of a false wall given the probability of a real wall (ROC curve) derived from the histograms. This shows that the OPD is an excellent indicator of the presence or absence of a wall. Figure 9 shows the ROC curves for other combinations of wall thickness and attenuation. The OPD is a good indicator of whether a wall is actually present except when the SNR of the reflection from the second wall is less than unity (20 dB for the first wall reflection). When the SNR is increased to 30 dB, then the second wall reflection becomes visible above the noise and the OPD can be used to determine the presence and absence of a wall.

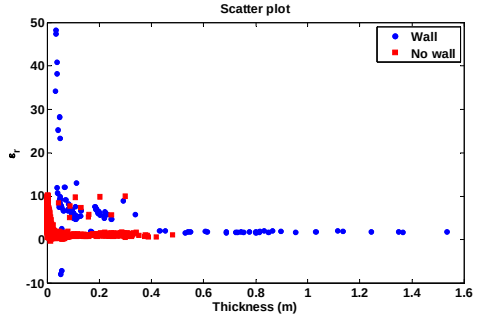

Figure 6. Scatter plot of thickness and relative permittivity of the second wall

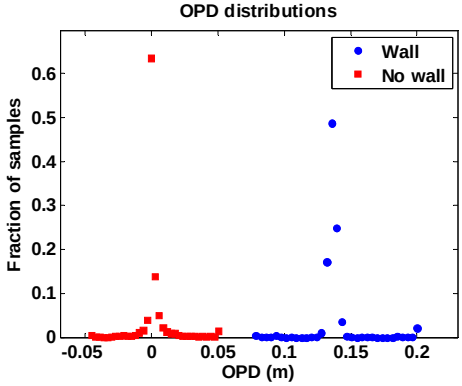

Figure 7. Histogram of OPD of the second wall.

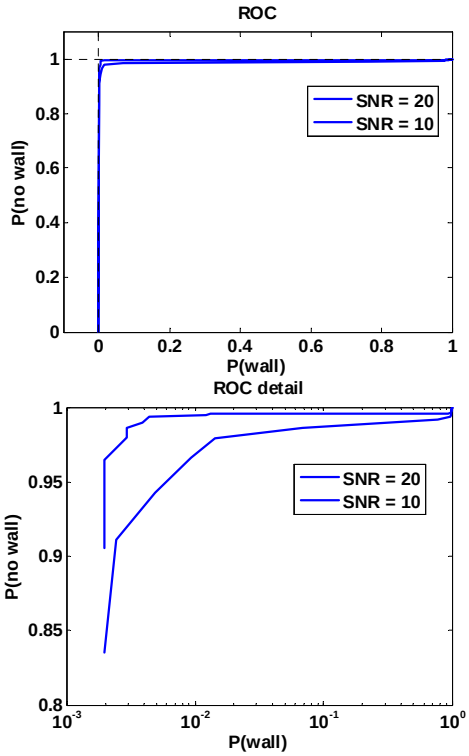

Figure 8. Probability of no wall as a function of the probability of a wall for SNR of 20 dB and 10 dB.

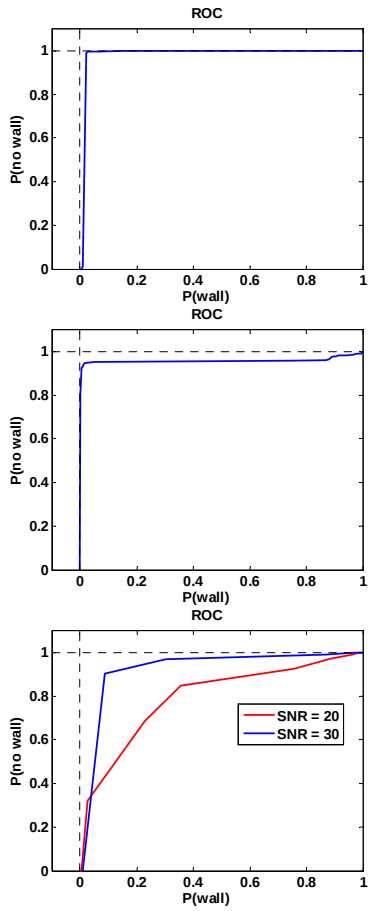

Figure 9. Curves of probability of no wall versus probability of a wall for SNR of 20 dB for thick, low attenuation walls (top); thin, high attenuation walls (center); and thick, high attenuation walls (bottom). The latter requires SNR >30 dB for good performance.

Summary

We have combined a gradient search algorithm with a non-gradient based genetic type search algorithm (particle swarm optimization) for extracting the properties of layers in a one-dimensional medium from simulated radar data. The PSO algorithm is used to calculate an initial guess for the gradient method that is close to the global minimum of the mean-square error. The hybrid algorithm shows good performance for a sequence of three walls of various thicknesses. It also was used to determine whether one could detect the presence or absence of a wall based on the optical path difference.

Acknowledgements

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

This work was performed under the auspices of the U.S. Department of Energy by University of California, Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.j

This work supported by DARPA.

References

1. W. C. Chew, *Waves and Fields in Inhomogeneous Media*, IEEE Press, Piscataway, NJ (1995)
2. M. Clerc, *Particle Swarm Optimization*, ISTE USA, Newport Beach, CA (2006)